CSCI 203 Place-out Project
milestone.pdf

The program first accepts a URL and creates a List of all words appearing on the web page. A helper function cleanContent() is called to remove all instances of punctuation marks, digits and stop-words (stored in stop-words.txt). Next, abstracting functionality of a word stemmer, a ModifiedStemmer Object is created. The program then invokes functions to create and sort according to frequency a List of tuples (word, frequency). Following is a sample program output with MAX_WORDS set to 5:

```
Please enter a URL
   http://www.eg.bucknell.edu/~csci203/placement/2016-spring/project/page1.html

Here is the dictionary of words on that page:
 {‘website’: 1, ‘love’: 1, ‘spam’: 10, ‘text’: 1, ‘university’: 1, ‘number’:
    1, ‘page’: 2, ‘bucknell’: 1, ‘example’: 1, ‘cloud’: 1}

Here is the text cloud for your web page:
spam (10)
page (2)
website (1)
university (1)
text (1)
```

For the first part of CSCI 203 programming project, I ensured that the program is robust. At first glance, coding stemContent() appeared as challenging, so I read a renowned paper on a stemming algorithm formulated by Martin Porter. Thereafter, I implemented the Porter Stemmer Class in a unique way, using only one-fourth the number of lines but accounting for almost all major suffixes.

To improve readability, I encoded the above class in modifiedStemmer.py file. On the other hand, the main program majorly follows functional programming paradigm.

List of Functions

- getContent(url)
  Returns as a List of splitted words on the user-entered website
- cleanContent(wordList, stopwords)
  Removes punctuation marks (string.punctuation) and digits using string.replace method, checks (and deletes) stopwords, returns as a List cleaned words
- stemContent(wordList)
  Stems words (usually, suffixed words) into root word, returns as a List stemmed words

- `filterContent(wordList)`
  Invokes helper functions to “clean” and “stem” words, returns as a `List` of pure words (no punctuations, stop-words or stemmed words)

- `findFrequency(finalWords)`
  Returns as a `Dict` of word-frequency pairs

- `findMostFrequentList(freqDict)`
  Finds defined number of most frequent words using `List.sort()`, adds word-frequency pairs to `freqStr` in required format, returns as a `List` tuples `(word, frequency)` in `DESC` order of frequency

- `ModifiedStemmer.update(self)`
  Updates local copy of word by removing already determined suffixes

- `ModifiedStemmer.isRestorable(self)`
  Returns `True` if ‘e’ is to be restored (e.g., structur would return `True`)

- `ModifiedStemmer.stemMajor(self)`
  Removes plurals and -ed or -ing or -er suffix

- `ModifiedStemmer.stemOther(self)`
  Removes major suffixes not accounted for by `stemMajor()`

- `ModifiedStemmer.stem(self, word)`
  Calls `stemMajor()` and `stemOther()` functions and returns the stemmed word to main program